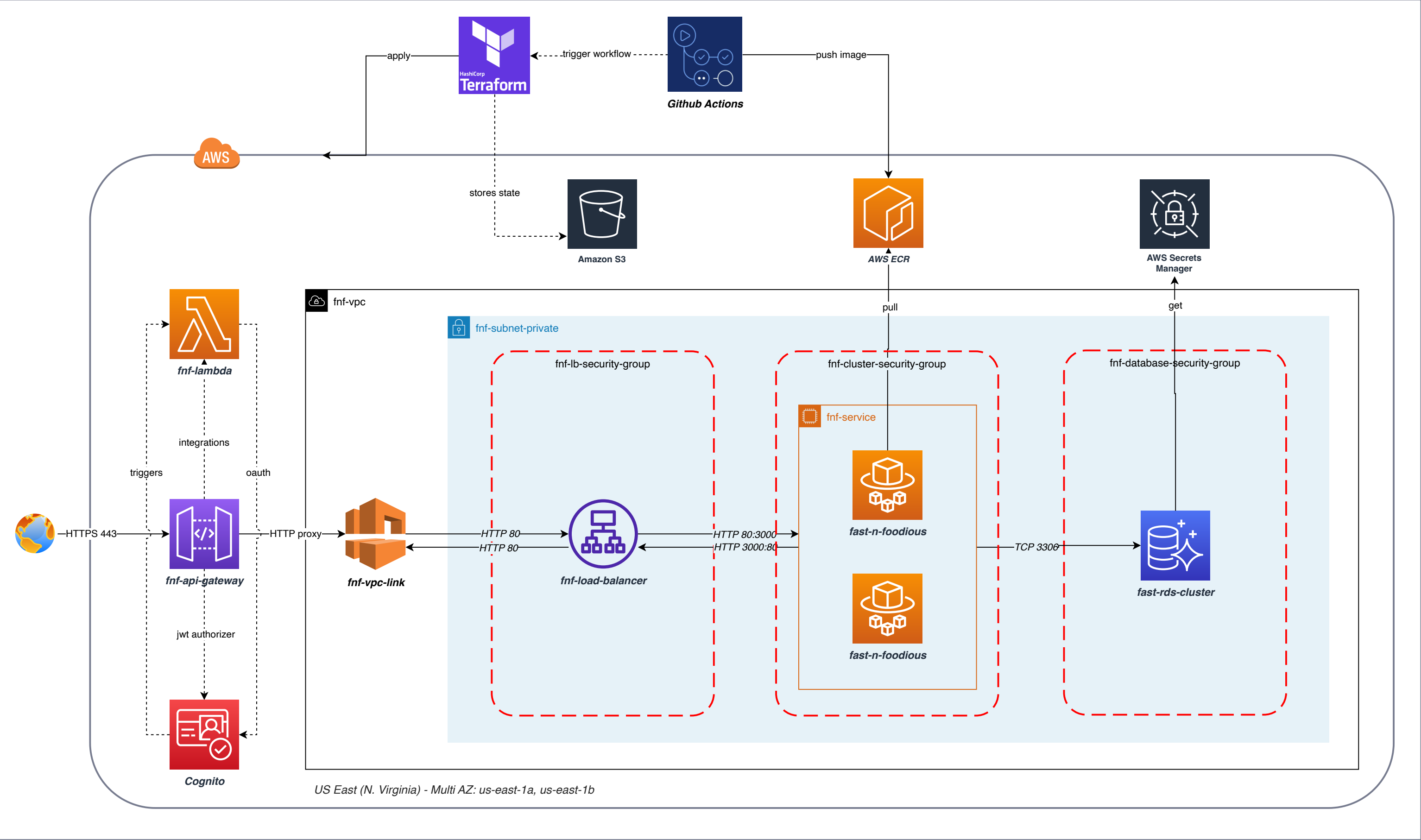
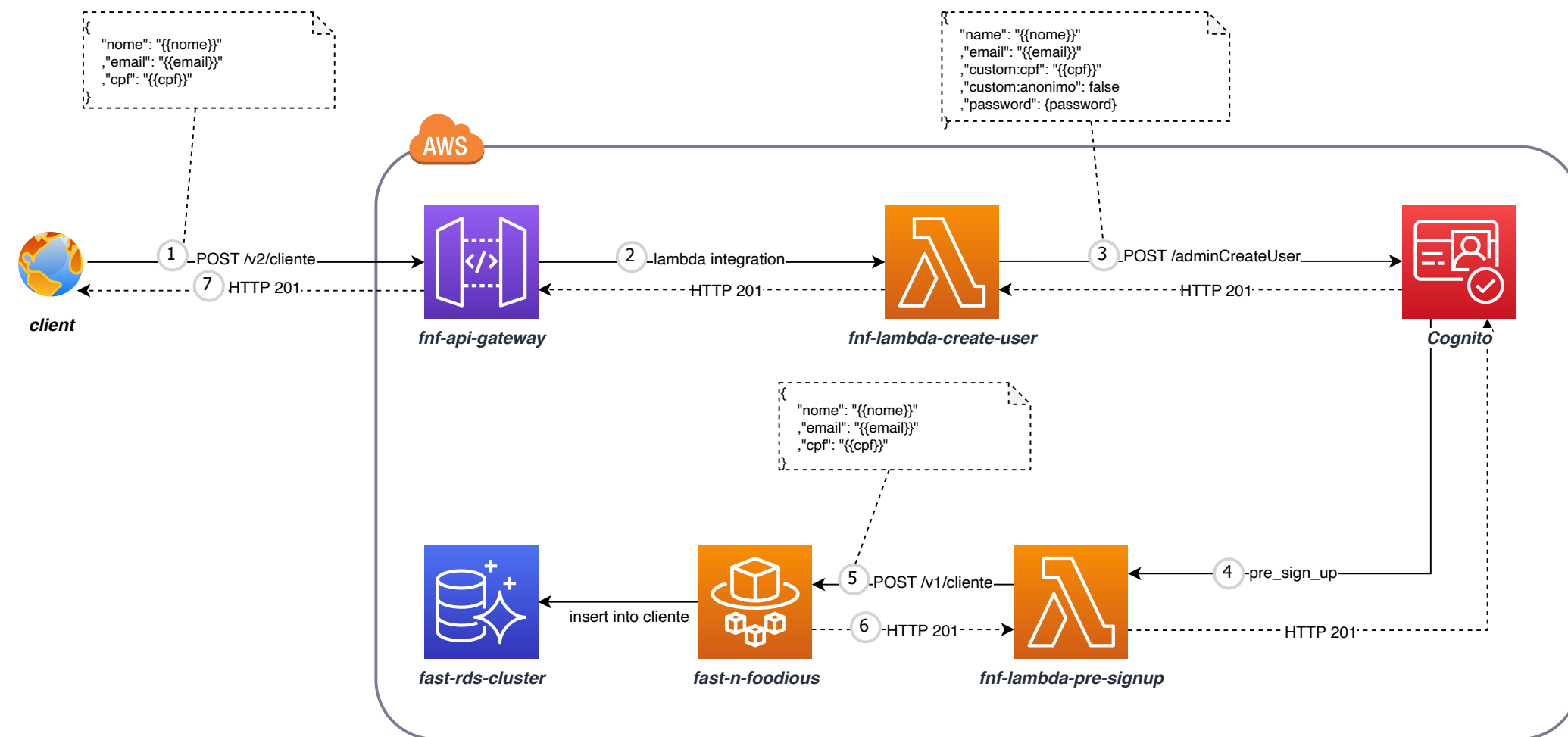


# Fast n' Foodious: Arquitetura Cloud AWS

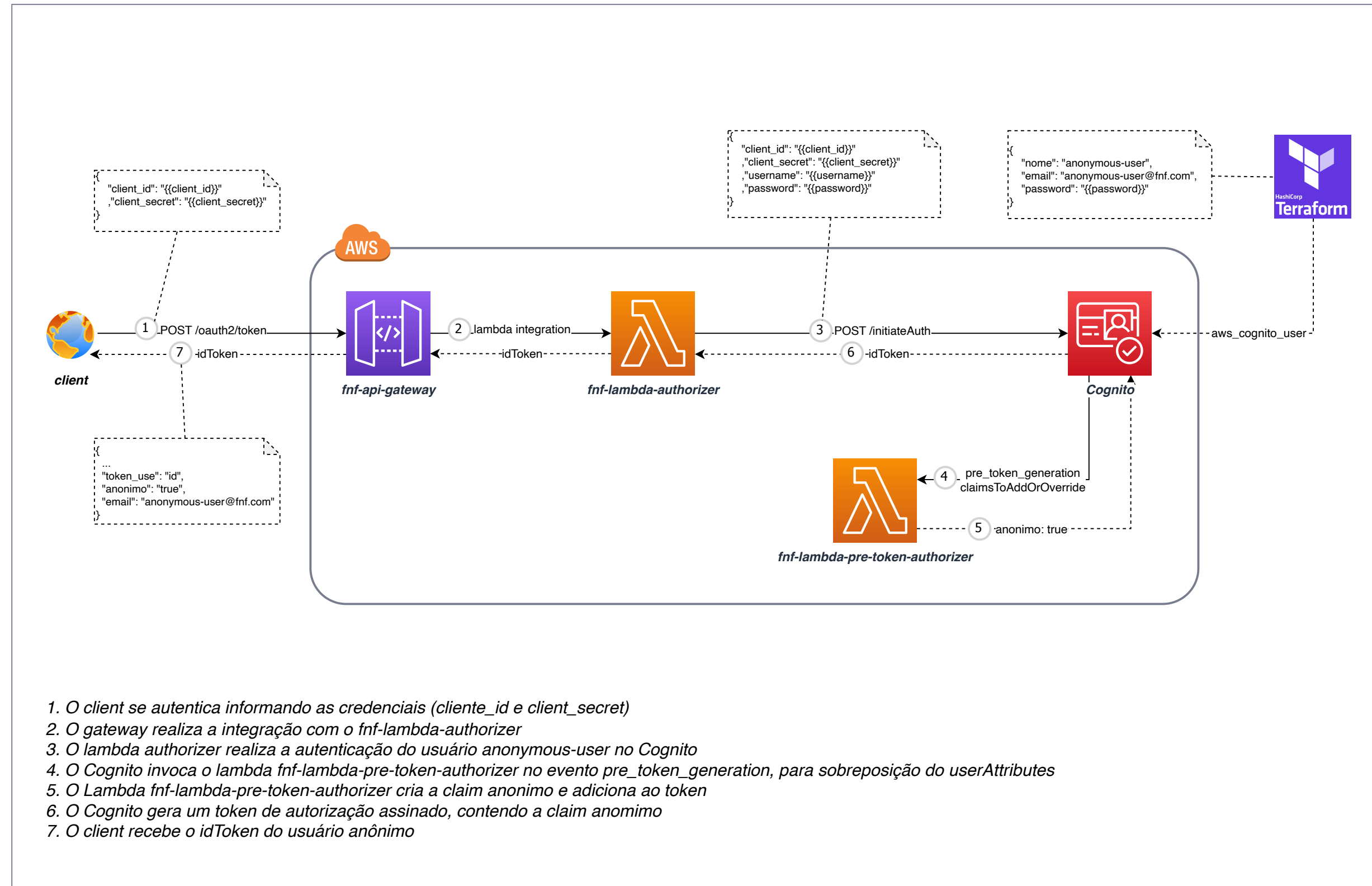


# Fast n' Foodious: Cadastro de Usuários

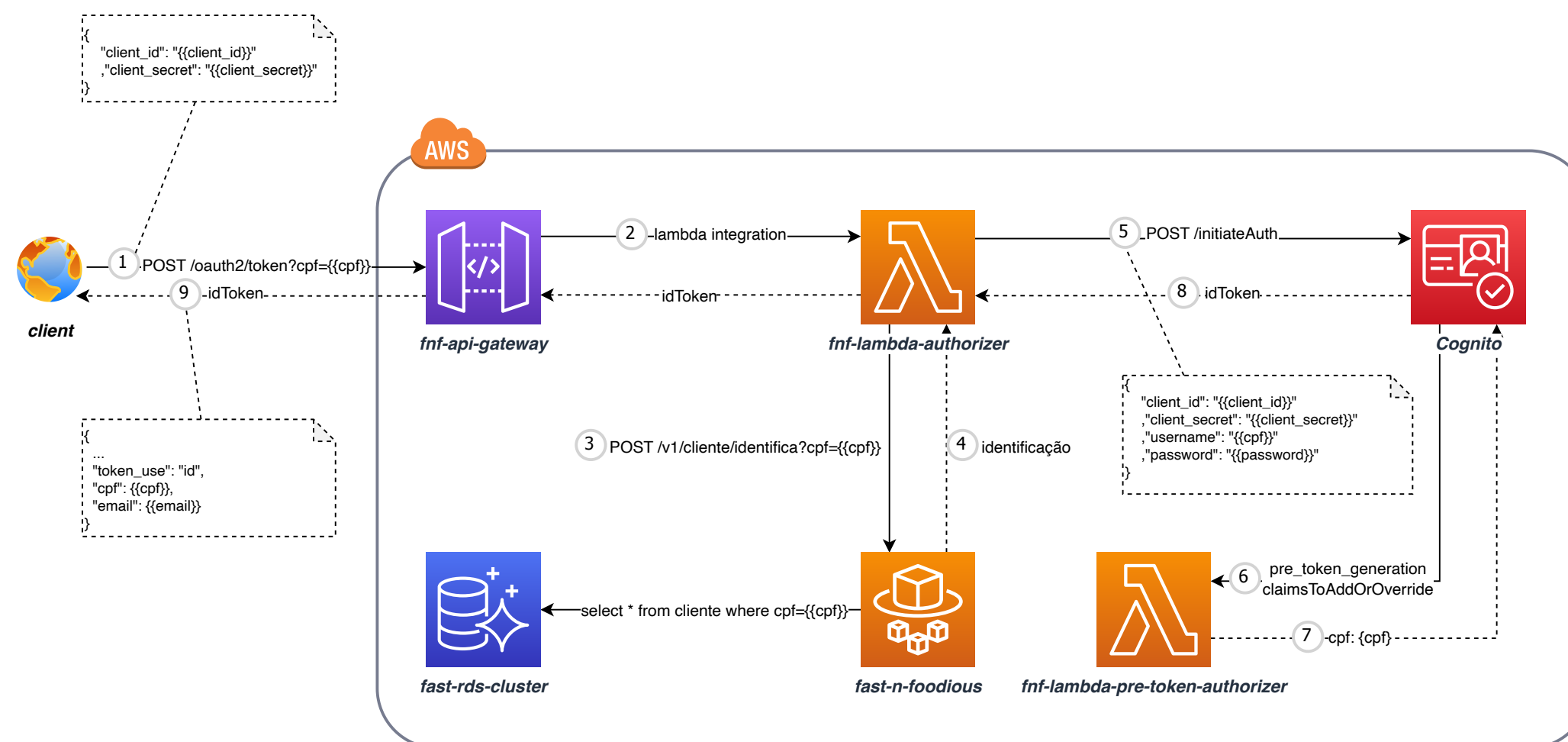


1. O client realiza o cadastro de usuário, passando os dados de nome, email, cpf
2. O gateway realiza a integração com o fnf-lambda-create-user
3. O lambda inicia o cadastro de novo usuário no Cognito
4. O Cognito invoca o lambda fnf-lambda-pre-singup, para inclusão de usuário na aplicação (cliente fast-n-foodious)
5. O lambda fnf-lambda-pre-singup realiza o cadastro do usuário na aplicação (cliente fast-n-foodious)
6. O sistema fast-n-foodious retorna HTTP 201, confirmando a operação de cadastro
7. O client recebe HTTP 201 como confirmação de usuário cadastrado

# Fast n' Foodious: Autenticação de Usuário Anônimo



# Fast n' Foodious: Autenticação de Usuário Identificado



1. O client se autentica informando as credenciais (cliente\_id, client\_secret, username e password)
2. O gateway realiza a integração com o fnf-lambda-authorizer
3. O lambda authorizer realiza a identificação do usuário (cliente fast-n-foodious)
4. O sistema fast-n-foodious retorna os dados do cliente identificado: de cpf, nome e email
5. O lambda authorizer realiza a autenticação do usuário identificado no Cognito
6. O Cognito invoca o lambda fnf-lambda-pre-token-authorizer, no evento pre\_token\_generation, para sobreposição do userAttributes
7. O lambda fnf-lambda-pre-token-authorizer cria a claim cpf e adiciona ao token
8. O Cognito gera um token de autorização assinado, contendo a claim cpf
9. O client recebe o idToken do usuário identificado